

CLAIMS

Please amend the claims as follows:

1. (Currently amended) A method of reducing spatial noise in an image having a plurality of pixels, the pixels being arranged in a plurality of (h) rows and (w) columns, comprising:

providing a set of three 64 bit registers, each register representing eight horizontally adjacent pixel values from one of three respective vertically adjacent image rows;

computing eight sets of directional high-pass values, one for each horizontal pixel position represented by said registers; and

computing directionally smoothed low pass pixel values by combining said high-pass values with image pixel values to produce directionally weighted sums; and

using the weighted sums to provide a reduced spatial noise image.

2. (Original) A method according to claim 1, wherein:

said three 64-bit registers include a first (up) register, a second (mid) register and a third (down) register, said first, second and third registers each representing pixel values for eight horizontally adjacent image pixels;

pixel values in said first (up) register represent image pixels horizontally aligned with and vertically adjacent to pixels represented by corresponding pixel values in said second (mid) register; and

pixel values in said second (mid) register represent image pixels horizontally aligned with and vertically adjacent to pixels represented by corresponding pixel values in said third (down) register.

3. (Original) A method according to claim 2, wherein:

said step of computing eight sets of directional high-pass values further comprises:

computing eight average values for corresponding first and third image row pixels in said first (up) and third (down) registers;

computing eight maxima and eight minima between said average values and corresponding second image row pixels in said second (mid) register;

determining eight high-pass absolute values by subtracting said minima from said maxima;

setting values of eight sign bytes such that if corresponding second row pixel values and minima are equal, a corresponding sign byte is set to all 1's, otherwise the corresponding sign byte is set to all 0's;

shifting the first (up) and third (down) registers for left diagonal alignment of pixel values;

calculating left diagonal high-pass absolute values and signs;

shifting the first (up) and third (down) registers for right diagonal alignment of pixel values;

calculating right diagonal high-pass absolute values and signs;

shifting said second (mid) register for horizontal alignment of pixel values; and
calculating horizontal high-pass absolute values and signs.

4. (Original) A method according to claim 2, wherein:

said step of computing directionally smoothed low pass pixel values further comprises:

clearing (zeroing) high-pass values to exclude from low pass filters;
clearing (zeroing) sign bytes for high pass values equal to zero;
computing low pass values for four upper bytes by summing four corresponding high-pass values and subtracting from four corresponding second image row pixel values in said second (mid) register; and
computing low pass values for four lower bytes by summing four corresponding high-pass values and subtracting from four corresponding second image row pixel values in said second (mid) register.

5. (Original) A method according to claim 2, wherein:

the first (up) register represents pixels $p(i-1, j+k)$ in a corresponding first (up) image row;

the second (mid) register represents pixels $p(i, j+k)$ in a corresponding second (mid) image row; and

the third (down) register represents pixels $p(i+1, j+k)$ in a corresponding third (down) image row.

6. (Original) A method according to claim 5, further comprising:

storing directional smoothing results for pixels $p(i, j+k)$ in place of $p(i-1, j+k)$.

7. (Currently amended) A method according to claim 5, further comprising:

computing scaled high-pass filters ($hp_filt(i, j+k, d)$) for an aligned vertical direction ($d=1$) for eight horizontally contiguous pixels simultaneously using the relationship:

$$hp_filt(i, j+k, d) = p(i, j+k) - (p(i-1, j+k) + p(i+1, j+k)) / 2 \quad k=[0, 7].$$

8. (Original) A method, according to claim 5, further comprising:

calculating low-pass filter values ($lp_MMX(i, j+k)$) in accordance with the relationship:

$$lp_MMX(i, j+k) = \left[4 \cdot p(i, j+k) - \sum_{d=1}^D hp_filt(i, j+k, d) \right] >> 2$$

$$k=[0, 7].$$

9. (Original) A method according to claim 5, further comprising:

determining a degree of directional smoothing for each pixel according to a number of directional high-pass values ($hp_filt(i, j+k, d)$) that have high-pass absolute values less than or equal to $|hp_{min}(i, j+k) + \Delta|$.

10. (Original) A method of reducing spatial noise in an image having a plurality of pixels, the pixels being arranged in a plurality of (h) rows and (w) columns, comprising:

providing a set of 64 bit registers for representing eight 8-bit pixel values, said registers including a first (up) register for representing eight horizontally adjacent pixel values from a first image row; a second (mid) register for representing eight horizontally adjacent pixel values from a second image row, said second image row pixels occurring vertically adjacent to and horizontally aligned with said first row pixels; and a third (down) register for representing eight horizontally adjacent pixel values from a third image row, said third image row pixels occurring vertically adjacent to and horizontally aligned with said second image row pixels;

loading said first (up), second (mid) and third (down) registers with pixel values from three vertically adjacent image row and saving previous register contents;

computing eight average values for corresponding first and third image row pixels in said first (up) and third (down) registers;

computing eight maxima and eight minima between said average values and corresponding second image row pixels in said second (mid) register;

determining eight high-pass absolute values by subtracting said minima from said maxima;

setting values of eight sign bytes such that if corresponding second row pixel values and minima are equal, a corresponding sign byte is set to all 1's, otherwise the corresponding sign byte is set to all 0's; shifting the first (up) and third (down) registers for left diagonal alignment of pixel values;

calculating left diagonal high-pass absolute values and signs;

shifting the first (up) and third (down) registers for right diagonal alignment of pixel values;

- calculating right diagonal high-pass absolute values and signs;
- shifting said second (mid) register for horizontal alignment of pixel values;
- calculating horizontal high-pass absolute values and signs;
- clearing (zeroing) high-pass values to exclude from low pass filters;
- clearing (zeroing) sign bytes for high pass values equal to zero;
- computing low pass values for four upper bytes by summing four corresponding high-pass values and subtracting from four corresponding second image row pixel values in said second (mid) register;
- computing low pass values for four lower bytes by summing four corresponding high-pass values and subtracting from four corresponding second image row pixel values in said second (mid) register; and
- packing and storing said low pass values into said first register.

11. (Currently amended) A method of reducing spatial noise in an image having a plurality of pixels, the pixels being arranged in a plurality of (h) rows and (w) columns, comprising:

- calculating a plurality of high-pass filter values for a number of pixels based upon surrounding pixel values in a 5x5 pixel area surrounding each pixel for which a high-pass filter value is to be calculated;

- determining directionality information for each from the high-pass filter values for each pixel;

- determining directional low-pass (smoothing) filter values for each pixel based upon the directionality information for each pixel; and

applying the low-pass filter values (smoothing) to each pixel according to pixel values in the 5x5 pixel area surrounding each pixel; and
displaying the spatially reduced noise image.

12. (Original) A method according to claim 11, wherein each high-pass filter calculation comprises a summation of products, each product being determined by multiplying each pixel value in a 5x5 pixel area surrounding each pixel by a corresponding high-pass filter coefficients.

13. (Original) A method according to claim 12, wherein directionality is determined by identifying a high-pass filter value with a highest absolute value above a threshold value.

14. (Original) A method according to claim 13, wherein the threshold value is determined by adding a constant to the value of the high-pass filter having a minimum absolute value.

15. (Original) A method according to claim 11, wherein the directional low pass filter values are determined to decrease the relative amount of smoothing applied in a direction indicated by the directionality information.

16. (Original) A method according to claim 11, wherein the number of low pass (smoothing) filters for each pixel value is equal to the number of corresponding high pass filters.

17. (Original) A method according to claim 16, wherein the number of high pass filters for each pixel value is four.

18. (Original) A method according to claim 17, wherein the filter directions are horizontal, vertical, diagonal right and diagonal left.

19. (Original) A method according to claim 16, wherein the number of high pass filters for each pixel value is eight.

20. (Original) A method according to claim 19, wherein the filter directions are, horizontal, vertical, diagonal right, diagonal left, greater diagonal right, greater diagonal left, lesser diagonal right and lesser diagonal left.